GPON FRAME DATABASE DESIGN

Martin Holik

Doctoral Degree Programme (1), FEEC BUT

E-mail: xholik11@stud.feec.vutbr.cz

Supervised by: Tomas Horvath

E-Mail: horvath@feec.vutbr.cz

Abstract: This paper deals with design of a database for capturing and saving frames transmitted in downstream direction over gigabit passive optical network (GPON). This kind of system uses optical transfer over passive optical networks and therefore the architecture of network is different than Ethernet technology. The useful payload is encapsulated in GEM frames by using the gigabit encapsulation method. In this article, we describe basic principles of passive optical networks and a database model for saving GPON frames in the downstream direction captured by the field programmable gate array (FPGA) card. Next, the fundamental scripts for analysing physical layer operation, admission and maintenance (PLOAM) messages are discussed. Finally, a developed Python tool for advanced analysis is shown.

Keywords: GPON, ONU activation process, PLOAM messages, Python, SQL database

1 INTRODUCTION

Over the last years, with increasing number of Internet users, the number of public services which are sharing between users is also increasing. The main purpose of these services are capabilities to gather, share and process information [1], [2]. Results of that development are requirements on the quality, stability and high-speed of Internet connection [3],[4].

These events have become the reason for increasing bit rate over transmission networks. That's why standard metallic lines are being replaced by active optical links in high-speed (backbone and access) networks. Also in the last mile connections the metallic lines are replaced by passive optical networks (PONs). Passive networks have shorter range than active networks but lower maintenance and operating costs have become their main advantage [5]. With the gradual development of this kind of optical networks, many technologies have also been developed that differ primarily in their transmission speed and network communication.

The definition of PONs is based on the recommendation of the International Telecommunication standardization section (ITU-T). Seems to be main problem with optical transmission because the manufacturers of optical devices don't have to follow the recommendations. The second disadvantage is networks analysis because for traffic analysis there exist only commercial analysing tools such as GPON Xpert™ or GPON-Doctor™. The application of these tools for monitoring network communication is determined by subsequent licensing policy and too high purchasing price. This limitation is critical for users, who in many cases want to perform a broader range of analyses.

2 PASSIVE OPTICAL NETWORK

Passive optical networks are deployed between end-users and access networks where they have replaced metallic-based networks. At present there by exist a few types of PONs which are distinguished termination on user's side. PONs consist of only passive devices which don´t require electric power. Active devices are used only as terminating units. The topology of networks is point-to-multipoint.

Transmission over GPON is based on time division multiplex (TDM). Each time slot contains one message for one specific optical network unit (ONU). In the downstream direction the traffic is transmitted as a broadcast. So evidently, each user side unit obtains also frames for other units on network.

3 DATABASE DESIGN

The transmission over a passive optical network is captured for detection purposes with own developed FPGA card. The main function of the card is to capture consecutive bits. The card is the processed by parsing software written in C#. This software compiles data from GPON frames and saves them for further processing and analysis in XML format. Captured and composed frames are saved into MS SQL for more advanced data operations. The advantage of using a database storage system is the ability to define the storage structure and set permissions for individual users. User-defined rules can be configured differently to enable security settings.

Future application of our project is based on selection of appropriate database design and structure. In Figure 1 database with name GPON for saving downstream traffic cauptured by FPGA card connected to optical network is shown. Captured GPON frames are decomposed into the GPON header and the useful payload which contain encapsulated GEM frames.

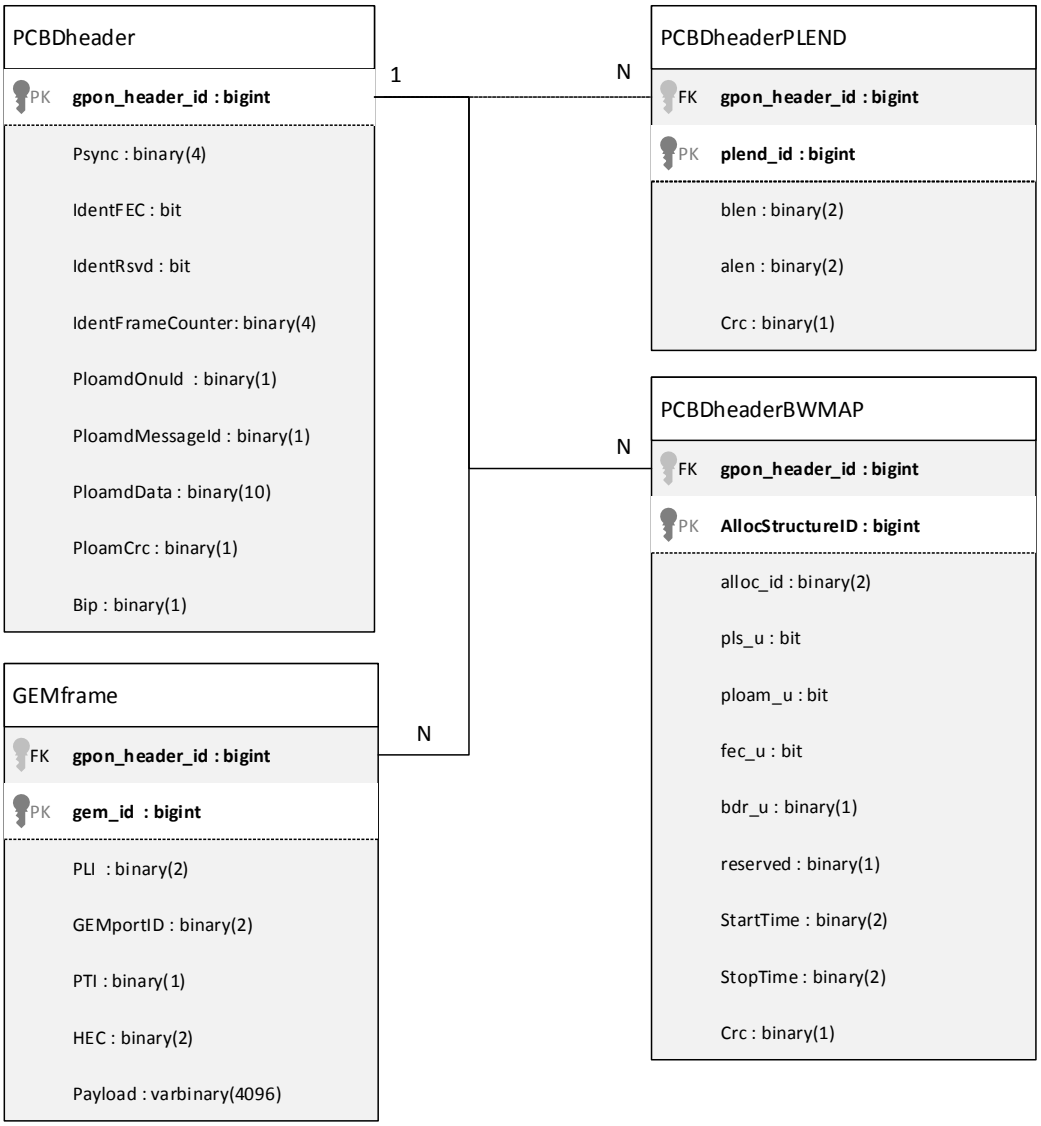

Figure 1: Database design

The process of saving is composed of several consecutive and dependent steps. Data are saved by using stored procedures on MS SQL server. In the first step, there are taken all fields from GPON header and they are stored in the database in the following order PSYNC, IDENT. After storing these fields are processed, in the second step controlling information from PLOAM fields. PLOAM messages are stored in the separate table and split with a unique key from previous table. At the end, there are also stored items from BWmap. The store process of this field is more complex than storing previous fields because BWmap is an array with the variable length. The field is stored by allocation structures. The allocation structure is N-bytes structure. Information about the off-site broadcasting is held in the one allocation structure. In figure 1 prepared storage for individual GEM frames is also shown. This storage is also disabled for saving field of GEM frame including user data.

The data in database is stored in raw form because it is better for the parser, which reads bits from the network and also for the next consecutive processing and analysis. For the database operation raw format is also better, but the main problem is with showing the contents to the user. Stored fields are displayed as bytes in hexadecimal format and for reading in decimal format internal or own written conversion function it must be used. The hexadecimal format is used when programming languages read data from the database. Loaded data are displayed in binary form but it is possible to make an easy conversion. The binary form is also better for the work in applications, for example in PLOAM messages, where certain bits are needed to detect the ONU-ID during the activation process.

	PSYNC	FEC	Rsvd	FrameCounter	OnuID	Ploam-ID	Ploam message name	PloamData	PloamCRC	BIP	GPON_ID
0	3064672736	1	0	850499514	0	*32	Undefined message!	b'\x01\xffY\xab \xaa\x00 \x00\x00 \x83'	96	0	1
1	3064672736	1	0	850499515	0	*32	Undefined message!	b'\x01\xffY\xab \xaa\x00 \x00\x00 \x83'	96	0	2
2	3064672736	1	0	850499516	0	*32	Undefined message!	b'\x01\xffY\xab \xaa\x00 \x00\x00 \x83'	96	0	3
3	3064672736	1	0	850500705	18	*30	Undefined message!	b'\x14\xff\x00 \x00\x00 \x00\x00 \x00\x00\x00'	96	0	4
4	3064672736	1	0	850500706	18	*30	Undefined message!	b'\x14\xff\x00 \x00\x00 \x00\x00 \x00\x00\x00'	96	0	5
5	3064672736	1	0	850500707	18	*30	Undefined message!	b'\x14\xff\x00 \x00\x00 \x00\x00 \x00\x00\x00'	96	0	6
6	3064672736	1	0	850539434	0	*32	Undefined message!	b'\x01\xffY\xab \xaa\x00 \x00\x00 \x83'	96	0	7

Figure 2: Output from basic script

Future analysis is possible on two independent levels MS SQL 2016 and Python programming language, but it should have been combined. The stored procedures and functions provide a very simple interface for processing at the database server. Then the user does not have to access to their source code. Inside the storage procedures and function are internally defined groups of commands

or database queries which on the something input returns some data operation. The typical result of a function is in most cases some value in decimal format for reading by the user. When the storage procedure is used an output table is created. These tables can be used in Python scripts. The Python script establishes the connection with database and calls stored procedures. When the procedure is executed, the Python script loads data from the table during SQL query, closes the database connection and runs its own code for data proceeding. Executed code continuously shows results in text format in the programming terminal and at the end when data is processed the results are exported to hypertext mark-up language (HTML) tables. For higher level presentation of the results, there are bootstrap cascading styles sheets (CSS), as can be seen in Figure 2. The analyses are performed with pre-defined input parameters. These parameters affect the analyses and also define if the analyses are performed for one specific ONU unit or for all ONUs.

The output of one of the scripts can be seen in Figure 2. There the script for filtering the ONU trigger process from the downstream traffic is used. After connecting each ONU to the network a sequence of defined PLOAM messages is initiated where by OLT and ONU determine the needed parameters for communication. This process is called activation process [6].

The GPON frames are processed so far at all levels. Downstream traffic captured by the FPGA card is processed by a parser application and stored in the database system. The Python script performs in the first phase a group of database calls via Transact-SQL and the server triggers the activation process. During the analysis the data is filtered and extracted into a special sequence. This sequence is in the next step saved in a separate table. The contents of the table are filtered PLOAM messages corresponding to the captured communication during the activation process. For downloading a table from the database by a Python script it is necessary to perform the next SQL call and then the script process in the analysis.

Analysed activation process is stored in an internal SQL database structure. During the analysis the PLOAM messages are checked with the expected data if the received messages are valid. When an inconsistency is detect by the script, the cells in the HTML table are marked red for better user orientation (see Figure 2). This script is from the package of scripts for GPON analysis. With this packages it is possible to follow all communication over the passive optical network.

Another script contains detection of all PLOAM messages. Captured messages are compared with PLOAM messages described in the recommendation of ITU-T G.984.3 and detected errors are graphically showed. The errors shown in Figure 2 were caused during debugging of database and parser software. The last but not a least we created very simple script which counts number of messages, detects ONUs and also creates a list of all ONUs ODN.

Scripts described above were used as the first concept for the analysis. For practical use their functionality will be implement as a more advanced and more user friendly application. This application will be developed as a web service based on the Flask framework. This tool allow the user the manage GPON analysis without knowledge the Python programming language and MS SQL database system which was necessarily for basic scripts. The application automatically manages database connection, runs required operations and captures and analyses the content on the website.

4 CONCLUSION

The proposed database is able to store GPON frames and run basic analysis on the SQL server or provide all captured data for later advanced processing. In the database there are saved disassembled GPON headers in binary format. In the simple analysis it is possible to detect PLOAM messages from downstream transmission. These messages are used for detecting activating and deactivating processes for one or multiple ONUs. In addition, the system can filter traffic for ONUs and even create a statistic of transmission. In combination with simple Python scripts it creates user-friendly output and provides managing analysis without knowledge of the database system.

The first version of the script, the system detected only parts of GPON header in the downstream direction, but now we detect all of the GPON frame and create user friendly tool for managing database by implementing basic scripts functionality into it.

All data from the network is obtained through our developed FPGA card. This is the main difference between our and commercial solutions. The advantage of our solution is its flexibility because by its modification, it can be used for many other purposes such as GPON safety testing. The parser software and database storage will be after completing downstream traffic analysing enhanced by upstream analysis.

The future goal is implementation of functionality for detecting and saving all GPON frames including encapsulated GEM frames transferring useful payload and also for detect frames in upstream direction. After successful decoding it will be possible to create methods of advanced analysis and implements to the graphical Python tool.

ACKNOWLEDGEMENT

The research described in this paper was financed by the grant of the Ministry of the Interior of the Czech Republic, Program of Security Research, VI20172020110, "Reduction of security threats at optical networks", by the National Sustainability Program under grant LO1401 and by the grant of the Brno University of Technology [FEKT-S-17-4184]. For the research, the infrastructure of the SIX Centre was used.

REFERENCES

[1] Sapolsky, H. M.; Crane, R. J.; Neuman, W. R.; Noam, E. M. *The Telecommunications Revolution: Past, Present and Future*; Routledge; Reissue edition: London, 2018; p. 217.

[2] Keshav, S. Paradoxes of Internet Architecture. *IEEE Internet Computing* **2018**, *22* (1), 96-102 DOI: 10.1109/MIC.2018.011581523.

[3] Houtsma, V.; van Veen, D.; Harstead, E. Recent Progress on Standardization of Next-Generation 25, 50, and 100G EPON. *Journal of Lightwave Technology* **2017**, *35* (6), 1228-1234 DOI: 10.1109/JLT.2016.2637825.

[4] Suzuki, N.; Miura, H.; Matsuda, K.; Matsumoto, R.; Motoshima, K. 100G to 1T based Coherent PON Technology. In *2017 European Conference on Optical Communication (ECOC)*; IEEE: Gothenburg, Sweden, 2017; pp 1-3.

[5] Mahloo, M.; Chen, J.; Wosinska, L. PON versus AON: Which is the best solution to offload core network by peer-to-peer traffic localization. *Optical Switching and Networking* **2015**, *15* (1), 1-9 DOI: 10.1016/j.osn.2014.03.001.

[6] Hood, D.; Trojer, E. *Gigabit-capable passive optical networks*; Wiley: Hoboken, c2012; p. 448.